CONFIDENTIAL.

slug

XU.LIGHT d44
93.4.23

XU•LT™:

SPECIFICATIONS FOR

XANADU LIGHT™

Note:

This series of pieces presents the logic for the whole XU•LT system.
The first portion is being submitted to the Hypertext 93 Conference.
The latter part is company confidential.

Submitted to the Hypertext 93 Conference (Seattle, November 1993).

TITLE: FILE FORMATS FOR OPEN METAMEDIA.

© 1993 Project Xanadu.
Note that "Xanadu," "Xanadu Light" and "XU•LT" are trademarks of Project Xanadu, Sausalito, California.

Author etc:

Theodor Holm Nelson
Xanadu™ World Publishing Repository Network
3020 Bridgeway #295
Sausalito CA 94965
Tel. 415/ 331-4422, Fax 332-**0136**

DISTRIBUTION LIST

Miura
Shigeru
Mark, Roger, XU guys
Doug
Brad Templeton
Dennis A.

SUMMARY

Many are concerned about how to annotate, link, quote among and combine published electronic objects.

We postulate here a new paradigm: METAMEDIA with TRANSPUBLICATION. That is, a publishing medium for all forms of data which allows arbitrary linkage and virtual inclusion among documents, with automatic royalty. We hope this will balance all contributions, ownership and points of view.

We believe this is a win-win solution, maintaining all intellectual property rights but bringing sweeping freedom of usage.

Arbitrary re-use is allowed, and thus anything may be seen in new contexts from new points of view;

nothing is misquoted (because the quoted material is drawn from the original at delivery time);
nothing is out of context (because the quoted material is drawn from the original at delivery time, and the original context may be stepped into by the user);
royalty is correctly apportioned;
credit is correctly apportioned.

Such a system can be accreted into an entire new hypermedia universe, a stable electronic literature, pluralistic and populist.
We discuss the technicalities of such a paradigm: fragment purchase and delivery, bivisible links, and the technicalities of addressing internal to the document and external to it.
We invite the use of this model for all data structures, and assert no proprietary rights to it.

INTRODUCTION

XANADU FALLOUT

What I will be discussing here is the general model we have developed on the Xanadu project, divorced from its technical and business particulars. It is time to do this, partly because everyone is studying these same issues, partly because we think we have the general solution, partly because the fate of the Xanadu project is uncertain, and partly because too few understand why we think we have the general solution.
No doubt the difficulty we've had explaining this to people has been commingled with various issues of paradigm, rivalry and business. The project's controversial history has made it hard to understand and has made many people unwilling to listen, partly because of the "religious" tinge of the project, and partly because we have promised secret methods of great efficiency that have so far failed to materialize. (One of these technical implementations, as well as our business model, is described in my book *Literary Machines*, available from Mindful Press, 3020 Bridgeway #295, Sausalito CA 94965.)

What matters is the idea; we will describe it here as separated out from that business venture, and separated from any of our technical implementations. Many variations are possible under this model. This will abstract its general properties, make the logic clearer, and elucidate the fundamental ideas we think necessary.

THE METAMEDIA PARADIGM

THE PREMISE: CONSTRUCTIVE METAMEDIA

Existing media objects are closed, designed as complete units. Yet we re-use their materials all the time. Short quotations are allowed; long quotations require immense organization. Documentary film and video producers must devote well over half their effort to the negotiation of rights. We propose to clean this up on a basis beneficial to everyone, with no one sacrificing anything, but allowing everyone to embed old parts in new documents without special arrangement.

By "metamedia" I mean the construction of new objects and documents-- meta-objects, meta-documents-- which may connect with the previous ones-- either by linkage, in various hyperstructures, or by virtual inclusion.

This is a proposal to create a new system of electronic publication, which requires both certain technical underpinnings and certain social arrangements.

PUBLISHING METAMEDIA: A COPYRIGHT SOLUTION

Metamedia publishing means that the new units build on the old, but the old keep their integrity. A key feature is *transpublication*, meaning that material virtually included in a new document pays automatic royalty to the original publisher. Participating documents can get royalty each time material is sent to a user.

This means that every time a document A with a transcluded quotation from B is delivered to a user, the user pays a royalty for the bytes of A to A's publisher, and a royalty for the bytes on transcluded material from B to B's publisher.

The key is the word "participation." Any publisher making fragments appropriately available with royalty is a player.

INTEGRITY OF ORIGINAL

The foremost priority is the integrity of the original document, with no intrusion on its owned document space. We will show how that is done.

SOCIAL SUBSTRATUM

A SYSTEM OF PUBLICATION LEGALISMS

Publication in legal sense means taking responsibility for the contents of what is published.
We will not go into that here.

A SYSTEM OF PERMISSIONS

There must be royalty arrangements and an understanding for the use of materials. We will not go into that here.

AN UNTESTED IDEAL

The metamedia paradigm is an untested ideal. We present it here to explain its technical practicality, which for some has seemed so doubtful as to render it beyond consideration.

TECHNICAL SUBSTRATUM

What we will discuss here is the technical substratum, below.

TECHNICAL REQUIREMENTS

There are a number of technical requirements for the metamedia world that are not familiar. I will enumerate them here and then go into more detail.

FRAGMENTARY DELIVERY

Contents must be requestable and usable by the arbitrary fragment.
This has many implications for structure

STRUCTURE WITHIN DOCUMENTS

We will be dealing with compound documents having unpredictable structure and an arbitrary mix of content types.
There will be many types, parts and structures.

We must allow for many data types to be included, and indeed must adapt existing file types to participate in these new virtual documents.

APPLICATIVE STRUCTURES

Must disembed codes and markings of all types
Contents must be made suitable to fragmentary delivery. This means disembedding codes and markings of all types, as well as header information and link ends.

STRUCTURE AMONG DOCUMENTS:

ALLOW CONNECTIONS TO REMOTE AND UNWITTING DOCUMENTS

We must be able to make all sorts of connections among documents. These include both transclusions (Contents distributed across document boundaries) and links of various types.

This includes connections to remote and unwitting documents.

IMPLICATIONS FOR ADDRESSING AND SEARCH SPACE

This has implications for the search space, and for addressing within it. In particular--

A document must have one linear address system;
The document owner may have no control of connections into the document.

PARALLEL PROTOCOL & IMPLICATIONS

FRAGMENTARY REQUEST AND DELIVERY

In this world, it must be possible to request and use contents by the arbitrarily selected fragment. (The whole document may be sent for as a series of fragments, or one big fragment.)

TRANSACTION MODEL

We are assuming that the server does not (and should not) keep session information as to where the user is and what the user is doing.

Thus the user simply keeps sending for pieces, arbitrary pieces, anonymously as far as server recordings are concerned.

Thus the transaction loop is typically as follows:

User. What connections are there?
System. The following connections...
User. Plz send fragment, here is payment.
Repeat.

How do you know what you got?

Each piece so sent must be independently usable. This has many implications for structure.

Every sent piece must be marked in some independent fashion, in parallel to the materials, rather than merged or embedded in it. This means the separate transmission of structural information, in a parallel marking system.

Thus there are two streams of information: the data and the metastream of identifiers.

INTERNALS & INTERNAL ADDRESS SPACE

In order to facilitate delivery of arbitrary fragments, and follow all connections bidirectionally, various technical features are necessary.

PARALLELIZATION OF INTERNALS

The entire system must be parallelized into a system where clean data is made parallel to its markings and links, rendering these markings and links applicative.

They must be in parallel structures that can allow any portions of clean data to be arbitrarily found and correctly identified.

This is necessary--

It is necessary for unimpeded scanning and search by arbitrary programs.

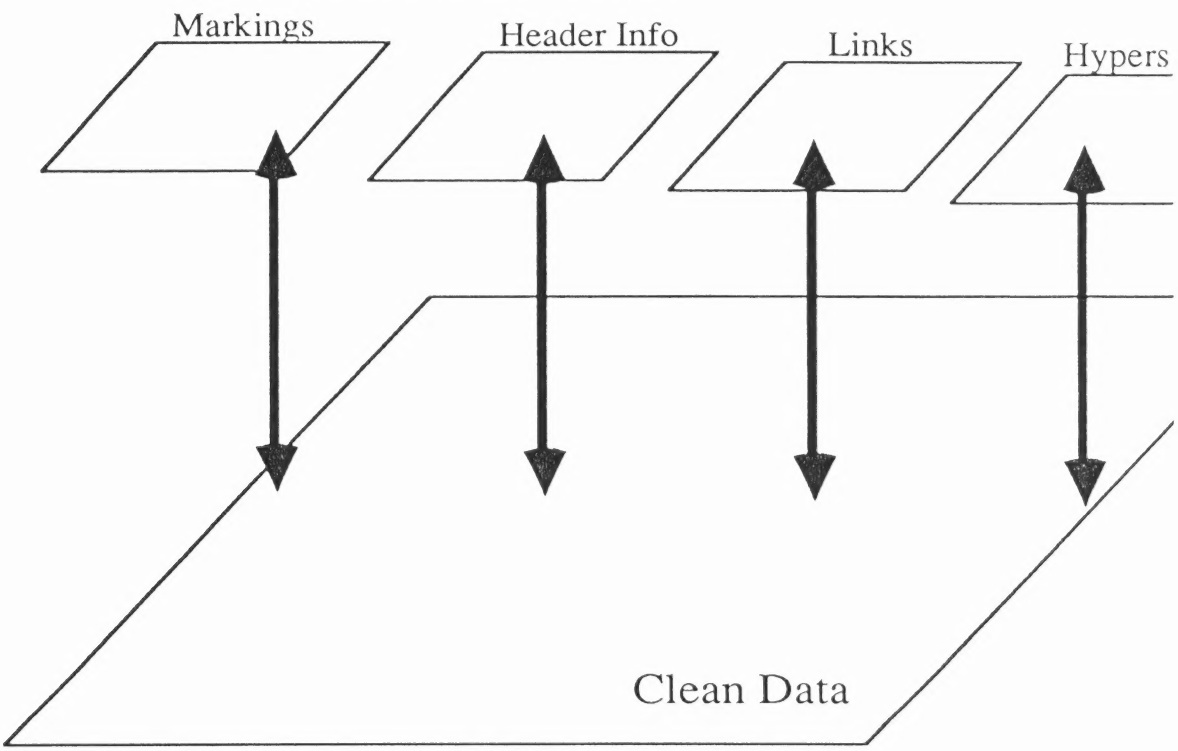
to allow delivery of independent fragments, sending them out to users for arbitrary use.

To allow external annotation without intruding on document space

To allow transclusive re-use for other purposes
To allow annotations conflicting with the original.
To allow many different annotations withut conflict.

HEADERS

In a conventional file there is usually a header, with global information and parameters about the file, and often embedded codes, as in word-processing files. In addition, the structure of hyperdocuments is sometimes represented in internals of the document. This is no longer workable.
The information associated with a file in a "header" can no longer be segregated at the front, but must be in some sense a parallel structure.



PARTS AND HYPERSTRUCTURE

Naturally, in tomorrow's document neither sequence nor any other structuring can be asssumed.
But for these purposes, the internal structures of a document-- its parts and connections, its systems of

sequence, jumps, overlay, etc.-- must be delineated in a separate box of structure, available in parallel to all the contained data.

Note that new structure may be applied to contents from outside.

EMBEDDED CODES, INTERNAL MARKINGS

Markings and attributes (such as paragraphs and italics) can no longer be marked with embedded codes. They must be displaced outward to parallel tables or the equivalent.

SINGLE INTERNAL ADDRESS SPACE

This parallelism is expressed as a single internal address space, referring in parallel to the data and all its markings, link-ends and structures. These are expressible in various tables or the equivalent.

We may illustrate them thusly:

ADDRESS SPACE (referred to by all others)

0 ---------- n

CONTENTS LIST

Tells where to find the individual parts of the document, whether locally stored or on remote servers.

There is no logical restriction as to where in the world they may be.

/ segment a addr. / seg.b addr / seg.c addr / ...

0 ---------- n

ROYALTY AND OWNERSHIP LIST

Tells the ownership and royalty charge on each segment, and where to send it.

(May be logically combined with Contents.)

/ segment a addr. / seg.b addr / seg.c addr / ...

0 ---------- n

Owned ATTRIBUTES, MARKINGS, OVERLAYS 1

--§-- ¡£∞----¶--•---º—º—ºª•---•¶-¶---†-------ø----√√---

0 ---------- n

Owned ATTRIBUTES, MARKINGS, OVERLAYS 2

ª---œ---´-----®---†--------¨---^---ø---π---å----ß----æ--Ω-

0 ---------- n

Owned ATTRIBUTES, MARKINGS, OVERLAYS 3

---≈--------ç------√∫---~µ---÷---∂ƒ---------©---˙---Δ°----¬

0 ---------- n

....

Owned LINK ENDSPANS 1 (overlap not shown)

<<----™->>--‹ fi ›‹ ›--≤--¢--≥-‹‹‹æ›››‹›‹---‹--‹‹-fi-››

LOCAL MAP

ROYALTY AND OWNERSHIP LIST

This is for coventional files having a header portion and a data portion.

It permits the system to refer to the header information and take from the contents information separately.

In these cases the contents portion will normally be included in the document's global address space but the header will not.

/ header / contents - - - - - ->

0 ----------/---------- a

Portion included in address space:

0 / xxxxxx /---------- a

which counts in global space 0 to n as:

i ..... j

EXTERNALS What happens Between Documents

The externals of the document make up what happens between documents. We may call these "connections;" in this context we have two types.

DISTRIBUTED CONTENTS-- (esp. TRANSCLUSIONS)

By transclusion I mean "inclusion across a document boundary." Other common names for it are virtual

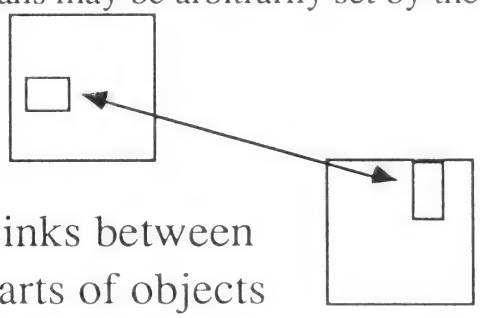
inclusion, inclusion by reference, logical inclusion, and "attach."

Transclusion should not be implemented as a link because it should be possible to link to these same materials in the new context; maintaining it as a distinct type makes this distinctly easier.

This has many advantages. One is that the contents of an owned document may be stored (permanently) elsewhere, and retrieved whenever needed; it also means that contents of an ***un***owned document may be a logical part of a new one. But best of all, it means we can have transpublication, discussed earlier.

LINKS

Links are a typed connection between bytes in one place and bytes in another. Links are between spans, not just insertion points. The spans may be arbitrarily set by the author.

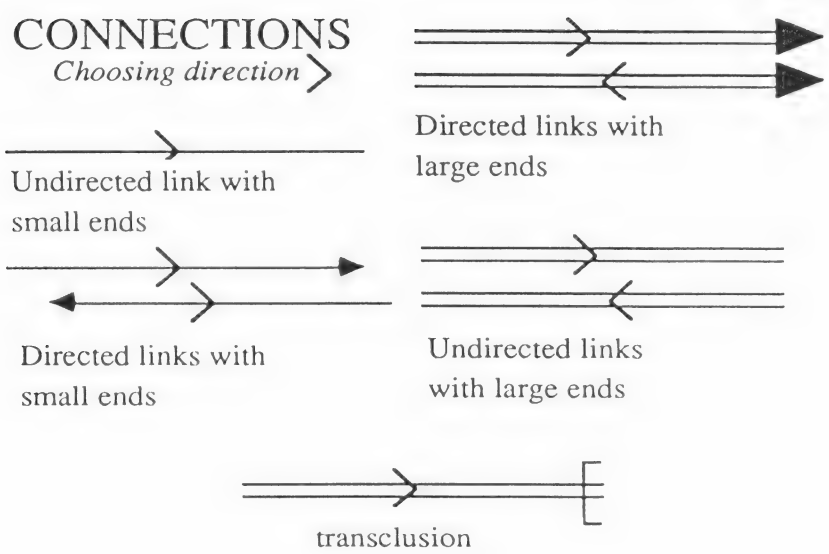


Links between parts of objects

(Possibly with different owners)

LINK DIRECTIONS

We must distinguish between the directionality of the link itself, and the direction in which it has been created or chosen by an author.



LINK TYPES

A link may be interpreted as a jump, a structural relationship, a literary relationship and more (see Appendix).

In the model we propose, there can be an arbitrary number of link types, expected to be in the thousands. Links also include applicative attributes and markings to the document.

EXTERNAL ADDRESSING USES INTERNAL ADDRESS SPACE

Connections from outside use the global address space of the document.

The following are maintained extermally by other publishers (as outpointers in the document's search space)

External ATTRIBUTES, MARKINGS, OVERLAYS 1

------Å-------Ï-------Ì-------Ó----------•-----Ò-------Ú-Æ

0 ---------- n

External ATTRIBUTES, MARKINGS, OVERLAYS 2

∂----*f*--√∫~--µ----®----ø---√√------°---©---Δ----æΩ--ç--

0 ---------- n

....

External LINK ENDSPANS (overlap not shown)

≤---------≥---≤---------≥---≤---------≥---≤---------≥-------

Externaly visible TRANSCLUSIONS

These are the externally visible ends of transclusions into another published document.

spans transcluded from this document:

B	C	A
/ --------- /	/ -------- /	/ --- /
and address in transcluding document	and address in transcluding document	and address in transcluding document

0 ---------- n

....

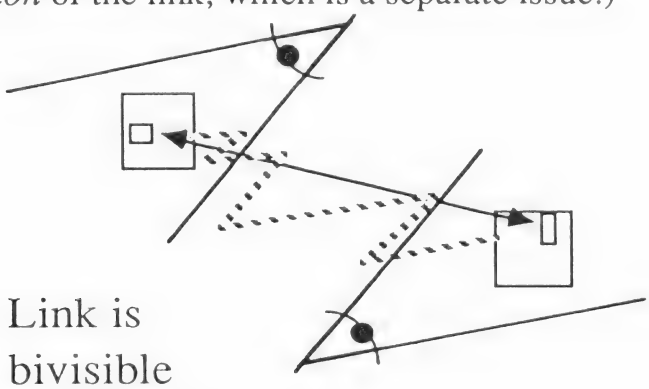
EXTERNAL ADDRESS SPACE

ALL CONNECTIONS OWNED

All connections are owned. This means that they are under the control of the originating or choosing author.

IMPLICATIONS OF BIVISIBILITY

In order for connections to be bivisible, they must be available both to the choosing document and the chosen document. (Note that this terminology is independent of the *direction* of the link, which is a separate issue.)

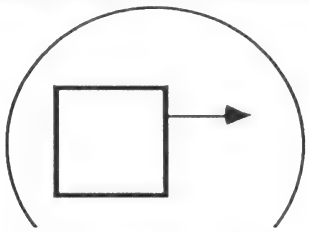


Link is bivisible (from both ends)

OUTPOINTERS AND INPOINTERS

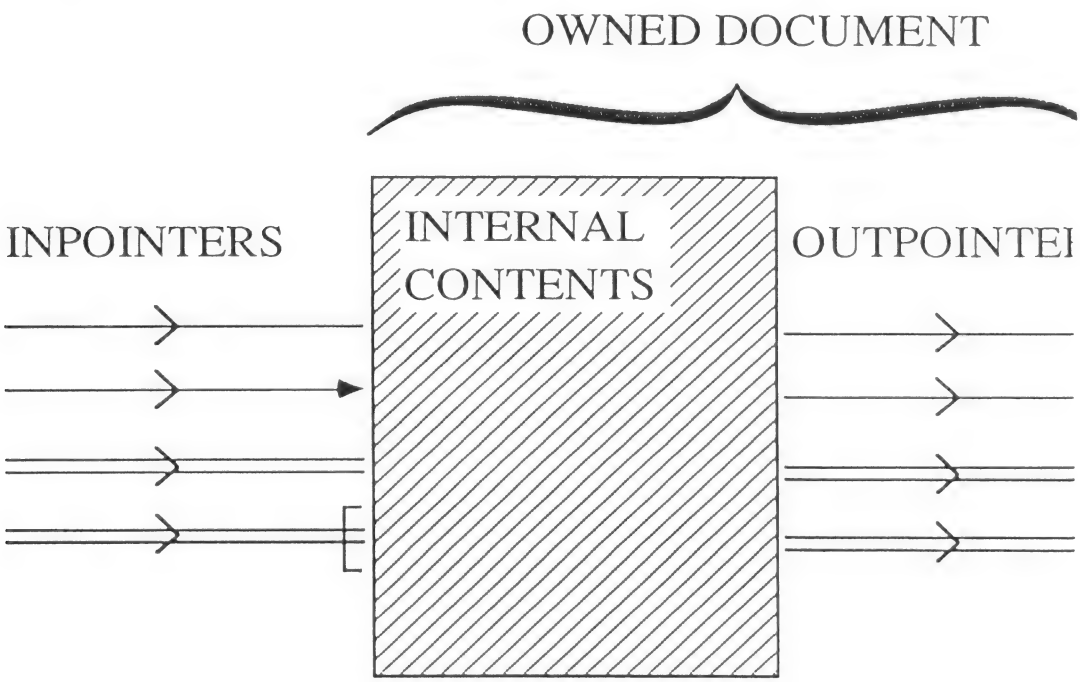
Another way to look at this is as inpointers and outpointers. From the choosing document, a connection is an outpointer.

But to the chosen document, it is an inpointer, more closely associated with the document chosen.



A documaent owns
its out-links.

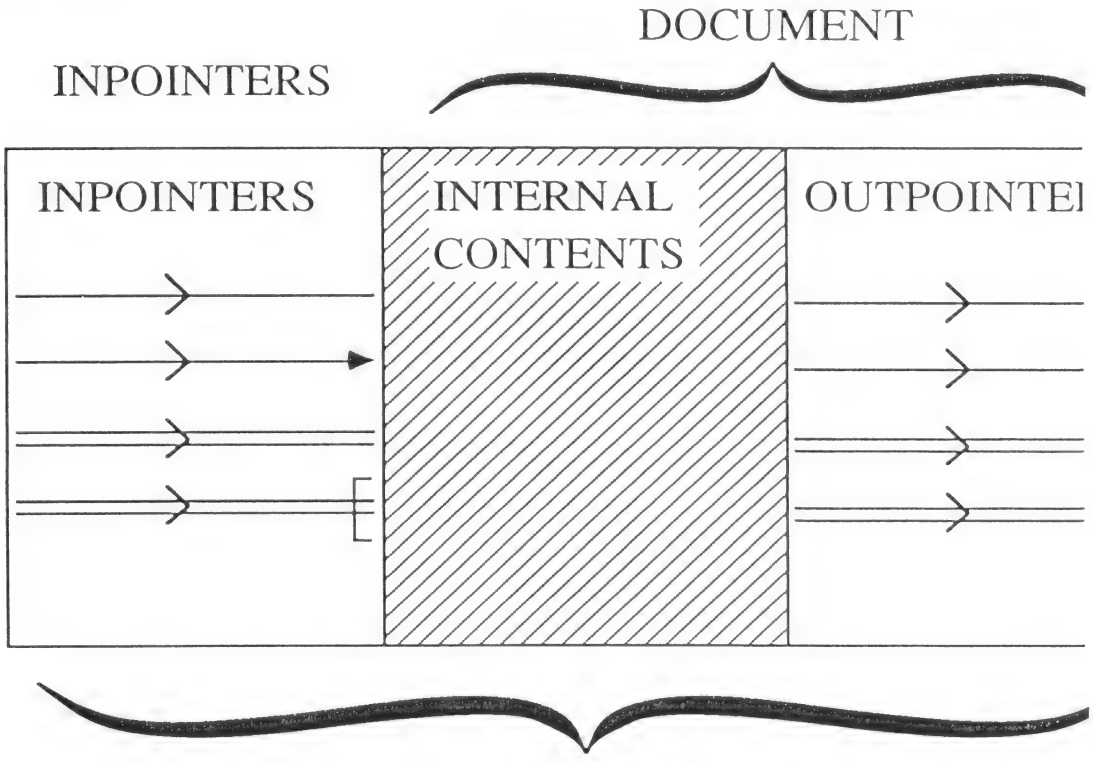
MY OUTPOINTERS ARE YOUR INPOINTERS



From the point of view of any document, there are out-links (which it owns) and in-links (which it does not). Thus everyone's out-link is someone's in-link and vice versa. (Note the analogy to what people *say*. Every person controls what he or she says, but does not control what others say to them.)

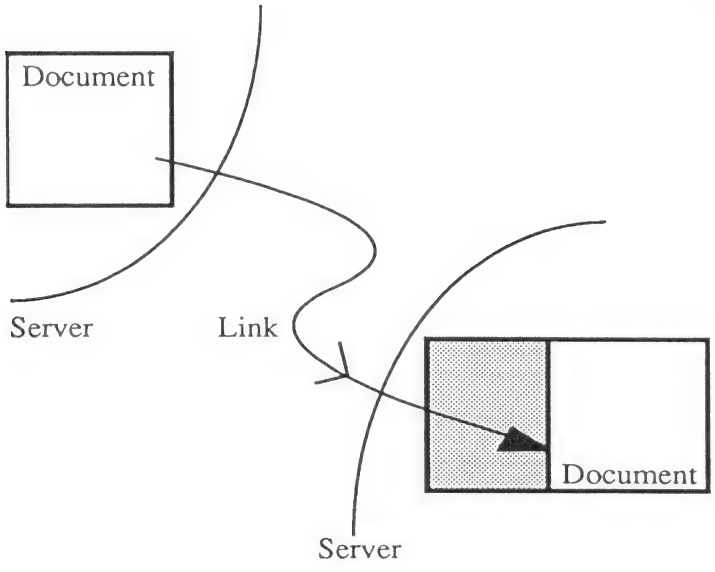
BARNACULAR MODEL

Every document keeps its integrity; the in-links do not affect its internal space. But the in-connectors are like barnacles, sticking to the outside.
They are, however, registered in an associated space-- the document's search space, which the document owner does not own.



Document's Local Search Space

We may think of a link as a harpoon, and the end which sticks-- the in-link-- as a barb which catches in the document's search space.



DOCUMENT NOT PUNCTURED

WHO OWNS THIS SPACE?

This extended search space is not owned by the document owners. Who then does own it? Who maintains it? Who pays for it? The question of who owns and maintains the search space of a given document is an administrative, political and business issue.

MAPPING TO ARBITRARY METHODS

There are obviously many technical ways to do these things. Ordinary database, suitable configured, can do it; object-oriented methods could be good; the real issue, of course, is the structure of the literature that results.

DISCUSSION, WRAPUP

The question is basically this: What world-tissue of documents are we to have?

TWO MODELS:

As I see it, there are two models widely on the table, which I think of as *puddings* and *shrapnel*.

By a "pudding" I mean a conventional, closed data object. This can be a file, a novel, a movie, a hypertext. No one may annotate it and no one may add to it.

By "shrapnel," I mean the shower of disconnected objects-- mostly puddings themselves-- which we get from GREP and GOFER functions. These may be exciting, but the context must be understood or researched.

I propose here a third model, which we may call *landscape*. In this model, all documents or objects form a great landscape, but their parts may be borrowed to put into specific frames-- transclusive publishing, with new contexts created freely by anyone.

I submit that in the long run it is context that we need, and more and more of it; and that we can architect such a landscape that is fair, free and clean.

I have brought up a couple of technical issues associated with the problem of how to build this kind of a world. The political issues are much greater, and such issues of freedom of speech and the press-- now transposed to the all-enveloping digital realm-- are among the most important we will ever face.

---------- **APPENDIX:**

SOME BASIC LINK TYPES (bivisible)

There has been considerable confusion caused by the term "bidirectional link." We must distinguish between several types of bidirectionality:

Symmetry. This is I think the truest sense of "bidirectional," meaning that a link is the same in either direction.

Two-way visibility, called here "bivisibility." This means you can see the link from either end, and presents special difficulties in multi-document implementation, discussed here.

Two-way passability. This means that you can *go* in either direction along the link. This is a moral issue. Many authors think it appropriate to restrict the reader. I believe this is evil.

In any case, here are a few link types. All should be made bivisible.

Untyped or Vanilla.

vanilla directed (eg HyperCard, implemented as a one-way irreversible jump)
vanilla undirected

Derived from text.
footnote
illustration
headline
sidebar
illustration
caption
marginal gloss

Having social/Literary meaning
comment
endorsement
disagreement

Structural.
sequence (of sections)
counterpart/correspondence

CONFIDENTIAL HEREAFTER-------
SPECIFICATIONS FOR
XANADU LIGHT™

PRELIMINARY BUSINESS ARRANGEMENT.

LEASED TO SERVICE PROVIDERS

The program will be available inexpensively to those wishing to publish their own materials or to furnish as public-access service providers.

The computer Bulletin Board community in the USA already has thousands of service providers; we will be marketing initially to this community.

A TWO-DONGLE SYSTEM

Program security will be maintained by two hardware locks, one on the search engine, one on the account/network system.

No Front End at first--

We can briefly postpone front-end programs--

Phase I Xanadu Light will have no front end.

At first, we will only have very simple commands to bring data.

The default front end at first will be keyboard commands asking for--

fragments of data
link information
data at the ends of links

OVERALL STRUCTURE OF PROGRAM

There will be TWO SEPARATE PROGRAMS.

Program I: Search and retrieval from the Xanadu Document

This will be on a relational database, implementing the model above (with some aspects not described here)
In Xanadu Light, we will be using generic (and thus low-cost and replaceable) database engines.
SQL routines will be the basic search routines.

Program II: Account management and network transmission

This will also work with a relational database, augmented with C routines.
This will do local accounting and account verification.
It will do network interchange with other Xanadu nodes, including:

Verification of non-local accounts
Sending for materials from other nodes
Supplying materials to other nodes
Sending billing tokens to other nodes

FILE STRUCTURE

First version: really simple file models.

A document is a directory's contents--

A compound document will be composed of standard files of whatever types, stored in a single directory.
Different data types are in different conventional files
Header information of a file type is addressed by type-specific routines

... as kept track of by the database engine.

The DBMS will keep track of the header and data addresses of these files, if applicable, and the respective positions of these materials in the overall virtual file.
Virtual count of actual data portion of each file is maintained by database engine

FURL AND UNFURL.

Some files do not fit the header/data model.
These are files which need to be unfurled for interpretation and the finding of addressable portions--

PostScript

fractal graphics
encryption
etc.

These programs will be run either at the server site (in which case the requested fragments will be delivered at additional cost) or the user site (in which case entire files must be sent). (We will be having discussions with Quarterdeck, Inc., whose proprietary and patented DESQVIEW program re-interprets the screen position of screen materials shipped out by running programs. This expertise will be particularly helpful in integrating UNFURL programs, running at the server, with delivery of materials from specific portions of the virtual document.

Evolutionary Parallelization of Data

Codes will be disembedded where possible;

eg a Microsoft Word document is converted to text with parallel specifiers

otherwise, they will be operated in Unfurl mode.

INTERNAL SEARCH SPACE: THE TABLES

CONTENTS OF DOCUMENT:

Location of each section

(May be on other servers or other networks)

File information where applicable: header, data

Furl information where applicable

Royalty rate per section

INTERNAL LINKS

Table of endpoints and types

INTERNAL MARKINGS

Table of endpoints and types

EXTERNAL SEARCH SPACE: THE TABLES

There will be a variety of tables for managing the incoming connectors, divided where necessary by priority and type.

EXTERNAL LINKS

Table of endpoints and types, and the pointing documents.

EXTERNAL MARKINGS

Table of endpoints and types, and the pointing documents.

TABLE OF INPOINTING TRANSCLUSIONS

What portions are used by what pointing documents.

WHO MAINTAINS WHAT SPACE? --

Responsibility and cost

Document owner maintains the Internal Space of a document

Pays for storage of document

Document owner pays costs of out-links

Costs of participating in a document's external space is borne by those linking from outside

Sending and maintaining harpoons & their storage

Search costs.

The Xanadu system maintains the External Space of a document.

However, the cost is the aggregate cost of all the out-links as paid for by the different publishers.

SESSION PROTOCOL

SIGNON

The user establishes ID and account.

The system verifies the account and sends the starting token.

The user receives the starting token.

(Further continuation numbers are supplied with authentication receipts, as described later.)

REQUESTS

Each user request begins with the continuation number, then asks for link information or a fragment.

REPLIES

The system replies with the requested information, and an authentication receipt (see below).

SIGNOFF

Billing is automatic.

The user may query the bill at any time.

BILLING

The user need never sign off.

However, a continuation token times out after a specific period, presumably less than an hour.

TRANSACTION THREAD SEQUENCE

1. Establish Account: Permissions, Billing

2. System. Here is first session token.

3. Request-delivery Loop: (more detailed than before).

User. Returning next session token.
What connections are there?
System. The following connections...
Here is next session token.
User. Returning next session token.
Plz send fragment.
System. Here is fragment.
Here is next session token.

AUTHENTICATION RECEIPTS

PARALLEL IDENTIFIERS to aid in filing and use

We have devised a simple, clean and parsimonious system of authentication codes to accompany delivered fragments. Each fragment, whether of content or structure or both, will be accompanied by a parallel authentication token.

This token serves both as an authenticator of the piece and as a receipt for its legitimate purchase.

It contains, for use in storage and identification of the fragment--

A SPECIFICATION OF THE CONTENTS OF THE FRAGMENT.

A HASH TO PROVE THAT THE CONTENTS ARE AUTHENTIC.

It further hashes in, for verification of the transaction and legitimacy of the copy--

A CONTINUATION CODE.

This continuation code is furnished in the user's next transmission, to continue the session thread.

It further contains, for verification of the transaction and legitimacy of the copy--

A TIME STAMP.
THE PRICE PAID FOR DELIVERY.
THE ROYALTY PAID.
THE CUSTOMER'S NAME.
This proves that the customer legitimately owns the copy.

Note that the system keeps no copy of the transaction, and the user's copy is the only proof that it occurred.

The system can prove the number and value of the materials sent, and these will balance the charges to users.

In a later version of the accounting software, there will be further transaction auditing methods-- all of which, however, will preserve the privacy of the transactions. They will prove things *about* the transactions, singly and in aggregate, but not keep a record of what the transactions *are*.